**Automation, Workflows and Agents**

**Level: Intermediate**

**Model Control Protocol (MCP): Mastering Claude API Interactions**

**2 days**

The Model Control Protocol (MCP) is the foundation for interacting with Claude AI models programmatically. This intensive, hands-on workshop equips developers with the knowledge and skills to leverage Claude's capabilities through well-designed API interactions.

You'll progress from basic API usage to implementing sophisticated patterns that solve real-world problems. Through practical exercises, code examples, and guided projects, you'll gain mastery of the MCP and learn how to integrate Claude's capabilities into your applications.

Claude models have transformed what's possible in natural language processing and AI assistance. By learning how to effectively prompt, control, and extend Claude through the MCP, you'll be able to build applications that leverage these powerful models while maintaining control over costs, performance, and user experience.

**Learning Outcomes**

By the end of this course, participants will be able to:

- Implement robust API interactions with Claude using the Anthropic Python client
- Apply effective prompt engineering techniques for consistent, high-quality responses
- Design and implement multi-turn conversations with proper context management
- Work with multimodal capabilities (text + images) to solve complex problems
- Build and deploy custom tools/functions for enhanced application capabilities
- Optimize token usage and implement cost-effective patterns
- Create retrieval-augmented generation (RAG) systems with Claude
- Implement proper error handling and fallback mechanisms
- Develop production-ready Claude integrations following best practices

**Course Outline**

**Prerequisites**

- Python programming experience
- Familiarity with REST APIs

- Laptop with Python 3.8+ installed
- Anthropic API key (obtain before workshop)

**Workshop Materials**

- Jupyter Notebook with complete code examples
- Reference documentation
- Solution templates for exercises

Workshop Structure

**Day 1: Fundamentals to Advanced Techniques**

**Morning Session (3 hours)**

**Hour 1: Introduction and Setup (9:00-10:00)**

- Claude capabilities and API overview
- Environment setup and authentication
- First API calls with the Anthropic Python client
- Hands-on: Sending basic prompts to Claude

**Hour 2: Effective Prompting (10:00-11:00)**

- Parameter tuning (temperature, max_tokens)
- System prompts and their impact
- Advanced prompting techniques:
    - Few-shot learning
    - Chain-of-thought reasoning
    - Structured outputs
- Hands-on: Optimizing prompts for specific tasks

**Hour 3: Managing Conversations (11:00-12:00)**

- Multi-turn conversation handling
- Context management strategies
- Streaming responses for real-time applications
- Hands-on: Building a simple conversational agent

**Lunch Break (12:00-1:00)**

**Afternoon Session (4 hours)**

**Hour 4: Multimodal Capabilities (1:00-2:00)**

- Working with images and text

- Document analysis patterns
- Multimodal prompt best practices
- Hands-on: Implementing a simple image analyzer

**Hour 5: Tool Use and Function Calling (2:00-3:00)**

- Tool schema definition
- Implementing single and multiple tools
- Handling tool responses
- Hands-on: Building a tool-enhanced assistant

**Hour 6: Performance Optimization (3:00-4:00)**

- Context window management
- Token optimization strategies
- Asynchronous processing patterns
- Hands-on: Processing batch requests efficiently

**Hour 7: Guided Project Work (4:00-5:00)**

- Start work on capstone mini-projects
- Instructor assistance and code review
- Implementation planning

**Day 2: Advanced Implementation and Production (Optional Extension)**

**Morning Session (3 hours)**

**Hour 1: Retrieval-Augmented Generation (9:00-10:00)**

- Vector embeddings basics
- Simple RAG implementation
- Improving relevance and accuracy
- Hands-on: Building a knowledge-based assistant

**Hour 2: Safety and Error Handling (10:00-11:00)**

- Content moderation approaches
- Robust error handling
- Rate limit management
- Hands-on: Implementing resilient Claude applications

**Hour 3: Production Considerations (11:00-12:00)**

- Model version migration

- Testing and evaluation
- Monitoring and logging
- Hands-on: Creating a version-agnostic interface

**Lunch Break (12:00-1:00)**

**Afternoon Session (4 hours)**

**Hour 4-6: Capstone Project Development (1:00-4:00)**

- Guided implementation of chosen project
- Apply multiple advanced techniques
- Individual assistance from instructor
- Code optimization and review

**Hour 7: Project Presentations and Conclusion (4:00-5:00)**

- Brief project demonstrations
- Best practices review
- Resources for continued learning
- Q&A and workshop wrap-up

**Capstone Mini-Project Options**

Participants will select one of the following projects to develop:

- Conversational document analyzer with RAG
- Multi-tool travel planning assistant
- Image-based product recommendation system
- Content moderation and summarization service

**Materials Provided**

- Comprehensive notebook with all code examples
- Starter templates for projects
- Reference cheat sheet for Claude API parameters
- Follow-up resources for continued learning

**Post-Workshop Support**

- 2-week access to workshop discussion forum
- Additional reference materials and examples
- Recorded solutions to all exercises

**Intended Audience**

This course is designed for software developers, data scientists, and technical professionals who want to integrate Claude AI capabilities into their applications. It's ideal for those looking to build AI-enhanced products, automate complex workflows, or develop custom AI assistants. The course provides practical approaches for both individual developers and teams implementing enterprise-grade AI solutions.

**Prerequisites**

Those attending this course should meet the following:

- Python programming experience
- Familiarity with REST APIs and JSON
- Basic understanding of AI and machine learning concepts
- Laptop with Python 3.8+ installed
- Anthropic API key (obtain before workshop)